AMENDMENTS TO THE DRAWINGS

The attached sheet of drawings includes new Fig. 5. Fig. 5 shows previously omitted elements 502, 502', 504 and 506.

Attachment: New Drawing Sheet

REMARKS

In response to the Office Action mailed on 12/05/2005, Applicant wishes to enter the following remarks for the Examiner's consideration. Applicant has amended claims 1, 10, 11 and 12. Claims 1-25 are pending in the application.

In the drawings, FIG. 5 has been added to show the first, second and third counters. Support for this new figure is provided by the pseudo-code listing from page 11, line 12, to page 12, line 8, and the associated description. This pseudo-code shows the use of a first counter (LEFT), a second counter (LEFT2) and a third counter (COUNT). The state machine and adder 506 performs the operations described in the pseudo-code and is disclosed in the specification on page 7, lines 6-7. Thus, no new material has been added in FIG. 5.

In the specification, the title of the invention has been amended to be more descriptive. Page one of the specification has been amended to update the status and provide the U.S. patent application numbers of the priority and cross-referenced applications. Also in the specification, FIG. 5 has been added to the Description of the Drawings section, and a description of FIG. 5 has been added to the Detailed Description section. This description closely follows the description of FIG. 2 and no new material has been added.

Rejection of claims under 35 USC §102

Claims 1-9 and 11-24 have been rejected under 35 USC §102 as being anticipated by Scales, III et al. (Scales) US Patent No. 6,202,130. Applicant respectfully traverses this rejection of the claims.

Claim 1 relates to a system for addressing a partitioned memory. The examiner suggests that, in figure 1, Scales shows a memory having a plurality of partitions. However, Scales column 2, lines 54-60 describes figure 1 as a data processor. Further, Scales explicitly states that the main memory is not shown in the figure. The main memory is not described as being partitioned and Scales provides no teaching of how to access data values in a partitioned memory.

In the Scales reference, the BLOCK STRIDE (108) is the stride between data values that are units of a vector (see column 1, line 66). A BLOCK is therefore a number of bytes that together form a unit of a vector. The BLOCK COUNT (106) is the span of a vector, that is, the number of blocks (data values) to be fetched by the operation (see column 5 lines 28-32). The BLOCK SIZE (104) of Scales is the size of each data value. Scales in column 5 lines 33-36 describes the STRIDE MODIFIER as the amount by which the stride is modified between each block of the data stream -- that is, the amount by which the stride is modified between each data value of the data stream. Applicant submits that this STRIDE MODIFIER is not equivalent to the SKIP value of claim 1. The SKIP value is an address modifier (rather than a stride modifier) that is related to the separation between partitions. Even if one were to consider Scales' BLOCK to be a partition consisting of a single data value (this is counter to the well known meaning of a memory partition and Scales does not teach this interpretation), the STRIDE MODIFIER is added to the stride value, whereas the SKIP value is added to the current address. This is described in the specification in the pseudo-code listing on page 9, line 25, for example.

The pseudo code performs the following steps:

read SPAN data values (each separated by STRIDE), skip SKIP data values,

read SPAN data values (each separated by STRIDE), skip SKIP data values,

etc.

Thus, for example the present invention allows the reading of a rectangular subarray. In particular, as shown in Figures 3 and 4, the stride value is the same in each partition – it is not modified, and the SKIP value is same between each partition.

Scales system performs the following steps:

read ONE data value, skip STRIDE data values,

read ONE data value, skip (STRIDE+STRIDE MODIFIER) data values,

*read ONE data value, skip (STRIDE+2*STRIDE MODIFIER) data values,*

etc

Thus, the number of data values skipped is modified at each read. In contrast, the present invention uses a fixed SKIP value. The Scales system can be used to access the diagonal elements of an array (column 2 lines 6-9), which requires varying skip values, but *cannot* be used to access a rectangular subarray, which requires a constant SKIP value and a constant STRIDE value.

In view of the above discussion, applicant submits that the STRIDE MODIFIER of the Scales reference is not equivalent to the SKIP value of claim 1. Therefore, Scales fails to teach, disclose, suggest or otherwise render obvious the use a of SKIP value, as defined in claim 1, to move between memory partitions.

Applicant has amended claim 1 to clarify this distinction. In the amended claim, the arithmetic unit is operable to calculate the address in the memory of a next element of the vector of elements from the address of a current element by adding a multiple of the SKIP value to the address of the current element if the next element is in a different partition to the current element and adding a multiple of the STRIDE value to the address of the current element otherwise. This amendment is supported in the specification by the pseudo code on pages 9 and 10 and by figures 3 and 4.

In light of the foregoing remarks, Applicant respectfully submits that the Scales reference fails to teach, disclose, suggest or otherwise render obvious the recitations of claim 1. Applicant thus respectfully requests that this basis of rejection of the claim be withdrawn and that a Notice of Allowance for claim 1 be mailed at the Examiner's earliest convenience.

Claims 2 and 3 depend from claim 1.

Claim 4. Referring to column 5 lines 42-45, Scales states that the values in the base address register 112 and offset register 114 are added together to form the effective address of the first element. In the present invention EA_START is the effective address of the first element. In claim 4, LEFT_START is not an address offset. Rather, it denotes the number of elements in the first partition. This has no influence on the starting address. Scales does not teach that the memory is partitioned, so has no equivalent to the value LEFT_START.

Claims 5-7 depend from claim 1.

Claim 8 calls for first and second counters. The examiner has identified elements 104 and 106 in Scales' figure 2 as counters. Applicant respectfully submits that interpretation is incorrect. Scales describes figure 2 as a graphical representation of a data stream touch (DST) load instruction (column 4. lines 60-63). The fields 104 and 106 are fixed data values in an instruction. In particular, element 104 is a BLOCK SIZE – this denotes the number of vector bytes to be received per element. Element 106 is a BLOCK COUNT that indicates the total number of blocks to be fetched. The data value BLOCK COUNT may be used to initialize a counter, but neither BLOCK COUNT nor BLOCK SIZE is a counter. Scales only describes a single counter -- the index *I* -- whereas Claim 8 calls for two counters.

Claim 9, which depends from claim 8, calls for two counters. The first counter, which is indicative of the number of elements remaining in a current memory partition, is reset when the end of the partition is reached. Scales only uses a single counter (the index *i*) which is indicative of the total number of elements read. In column 7, line 21, Scales describes how the complete DST hardware is reset to stop the fetch operation. Resetting the first counter of the present invention occurs when moving from one memory partition to the next – the fetch operation is not terminated. This is described in the specification with reference to figures 3 and 4. In figure 3, for example, the first counter (the LEFT value) is reset from to 5 when a new memory partition is reached.

Claim 11. As described above with reference to claim 1, the Scales reference does not teach the use a SKIP value. Applicant has amended claim 11 to clarify the distinction between the SKIP value of claim 11 and the STRIDE MODIFIER of Scales. In the amended claim 11, the input arithmetic unit is operable to calculate the address in the memory of a next element of the vector of elements from the address of a current element by adding a multiple of the SKIP value to the address of the current element if the next element is in a different partition to the current element and adding a multiple of the STRIDE value to the address of the current element otherwise. This amendment is supported in the specification by the pseudo code on pages 9 and 10 and by figures 3 and 4.

Claim 12, which depends from claim 11, has been amended in light of the amendment to claim 11.

Claims 13 and 14 depend from claim 11.

Claim 15 relates to system for accessing a partitioned memory. As described with reference to claim 1 above, Scales does not provide a means to address a partitioned memory. Claim 15 calls for first and second counters. The examiner has identified elements 104 and 106 in Figure 2 as counters. Applicant respectfully submits that interpretation is incorrect. Scales describes Figure 2 as a graphical representation of a data stream touch (DST) load instruction (column 4. lines 60-63). The fields 104 and 106 are data values. In particular, 104 is a BLOCK SIZE – this denotes the number of vector bytes to be received per element. The BLOCK COUNT 106 indicates the total number of blocks to be fetched. The data values in these fields may be used to initialize counters, but they are clearly not counters. Scales only describes a single counter -- the index *i*. Scales does not teach the use of a second counter that indicates the number of elements remaining in a memory partition. Further, Scales does not teach resetting a second counter if no elements remain in the current partition. Scales only teaches resetting the hardware if all of the vector elements have been accessed.

Claims 16-24 depend from claim 15. Although additional arguments could be made for the patentability of these claims, such arguments are deemed to be unnecessary in view of the above remarks with respect to claim 15.

In light of the foregoing amendment and remarks, Applicant respectfully submits that the Scales reference does not teach, suggest, disclose or otherwise anticipate the recitations of claims 1-9 and 11-24. Applicant thus respectfully requests that this basis of rejection of the claims be withdrawn and that a Notice of Allowance for these claims be mailed at the Examiner's earliest convenience.

Allowable Subject Matter

Claim 10 has been rewritten in independent form including all of the limitations of claim 1. Applicant requests allowance of this claim.

Claim 25 has been allowed.

In light of the foregoing amendments and explanations, applicant submits that all rejections of claims 1-9 and 11-24 have been overcome. The scope of the amended claims 1, 10, 11 and12 is substantially the same with implicit meaning now made explicit. Allowance of claims 1-25 is therefore respectfully requested at the Examiner's earliest convenience. Although additional arguments could be made for the patentability of each of the claims, such arguments are believed unnecessary in view of the above discussion. The undersigned wishes to make it clear that not making such arguments at this time should not be construed as a concession or admission to any statement in the Office Action.

Please contact the undersigned if you have any questions regarding this application.

Respectfully submitted,

Renee' Michelle Leveque

Leveque Intellectual Property Law, P.C.
Reg. No. 36,193
221 East Church Street
Frederick, Maryland 21701
301-668-3073
Attorney for Applicant(s)